IN THE CLAIMS

Please amend the claims as follows:

Claim 1 (Original): A lossless coding method for a digital signal in a floating-point format, comprising:

(a) a step of converting a first digital signal sample in the floating-point format into a second digital signal sample in an integer format;

(b) a step of losslessly compressing a sequence of said second digital signal samples in the integer format to produce a first code sequence;

(c) a step of producing a difference signal that corresponds to the difference between said second digital signal sample in the integer format and said first digital signal sequence in the floating-point format; and

(d) a step of outputting said first code sequence and difference information corresponding to said difference signal as a coding result.

Claim 2 (Original): A lossless coding method for a digital signal in a floating-point format according to Claim 1, in which said step (d) comprises a step of losslessly compressing said difference signal to produce a second code sequence and a step of outputting said second code sequence as the difference information corresponding to said difference signal.

Claim 3 (Original): A lossless coding method for a digital signal in a floating-point format according to Claim 1 or 2, in which said step (c) comprises a step of converting said second digital signal sample into a third digital signal sample in the floating-point format and a step of producing the difference between said first digital signal sample and said third digital signal sample as said difference signal.

Claim 4 (Original): A lossless coding method for a digital signal in a floating-point format according to Claim 1, in which said step (a) comprises:

(a-1) a step of detecting, for each of blocks each containing a plurality of first digital signal samples in the floating-point format, the maximum value of the exponents of the first digital signal samples;

(a-2) a step of adjusting the values of the exponents of the first digital signal samples in the relevant block by an adjustment value determined based on the detected maximum value so that the maximum of the numbers of digits of integer values converted from the first digital signal samples in the relevant block be a predetermined value;

(a-3) a step of converting the first digital signal samples in the floating-point format in the relevant block whose exponents have been adjusted into the integer format; and

(a-4) a step of coding adjustment information corresponding to said adjustment value to produce an auxiliary code, and

said auxiliary code is also output in said step (d).

Claim 5 (Original): A lossless coding method for a digital signal in a floating-point format according to Claim 1 or 4, further comprising:

a step of down-sampling said first digital signal sample or said second digital signal sample to make the sampling frequency of the digital signal sample used for producing said first code sequence lower than the sampling frequency of said first digital signal sample; and

a step of up-sampling the digital signal sample used for producing said first code sequence to convert the sampling frequency thereof to the same sampling frequency as said first digital signal sample, thereby producing a digital signal sample used for producing said difference signal.

Claim 6 (Original): A lossless coding method for a digital signal in a floating-point format according to Claim 1 or 4, further comprising:

a step of converting the quantization precision of each first digital signal sample or each second digital signal sample to make the quantization precision of the digital signal sample used for producing said first code sequence lower than the quantization precision of said first digital signal sample; and

a step of raising the quantization precision of the digital signal sample used for producing said first code sequence to the quantization precision of said first digital signal sample, thereby producing a digital signal sample used for producing said difference signal.

Claim 7 (Original): A decoding method for a digital signal in a floating-point format, comprising:

(a) a step of decoding and expanding a first code sequence to produce a first digital signal sample in an integer format;

(b) a step of producing a difference signal based on difference information;

(c) a step of converting said first digital signal sample in the integer format into a second digital signal sample in the floating-point format; and

(d) a step of combining said first digital signal sample in the floating-point format and said difference signal to produce a third digital signal sample in the floating-point format.

Claim 8 (Original): A decoding method for a digital signal in a floating-point format according to Claim 7,

in which said step (b) comprises a step of decoding and expanding said difference information to produce said difference signal in the floating-point format.

Claim 9 (Original): A decoding method for a digital signal in a floating-point format according to Claim 7 or 8, further comprising:

a step of decoding an auxiliary code to produce adjustment information; and

a step of adjusting the digit of said first digital signal sample in the integer format or said combined signal based on said adjustment information.

Claim 10 (Original): A decoding method for a digital signal in a floating-point format according to Claim 7 or 8, further comprising:

a step of up-sampling said first digital signal sample in the integer format to make the sampling frequency thereof equal to the sampling frequency of said difference signal, before converting said first digital sample into said second digital signal sample in the floating-point format.

Claim 11 (Currently Amended): A decoding method for a digital signal in a floating-point format according to Claim 7 or ~~10~~ 8, further comprising:

a step of converting the quantization precision of each first digital sample in the integer format to the quantization precision of said difference signal, before converting said first digital sample into said second digital signal sample in the floating-point format.

Claim 12 (Original): A lossless coder for a digital signal in a floating-point format, comprising:

an integer formatting part that converts an input first digital signal sample in a floating-point format into a second digital signal sample in an integer format;

a first compressing part that losslessly compresses a sequence of second digital signal samples in the integer format to produce a first code sequence; and

a difference producing part that produces a difference signal corresponding to the difference between said second digital signal sample in the integer format and said first digital signal sample in the floating-point format,

in which said first code sequence and difference information corresponding to said difference signal are output as a coding result.

Claim 13 (Original): A lossless coder for a digital signal in a floating-point format according to Claim 12, further comprising:

a second compressing part that losslessly compresses said difference signal to produce a second code sequence and outputs the second code sequence as said difference information corresponding to said difference signal.

Claim 14 (Original): A lossless coder for a digital signal in a floating-point format according to Claim 12,

in which said difference producing part comprises a floating-point formatting part that converts said second digital signal sample into a third digital signal sample in the floating-point format and a subtracter that produces the difference between said first digital signal sample in the floating-point format and said third digital signal sample as said difference signal.

Claim 15 (Original): A lossless coder for a digital signal in a floating-point format according to Claim 12, further comprising:

a digit adjusting part that detects, for each of blocks each containing a plurality of first digital signal samples in the floating-point format, the maximum value of the exponents of the first digital signal samples and adjusts the values of the exponents of the first digital signal samples in the relevant block by an adjustment value determined based on the detected maximum value so that the maximum of the numbers of digits of integer values converted from the first digital signal samples in the relevant block be a predetermined value; and

an auxiliary coding part that codes adjustment information corresponding to said adjustment value to produce an auxiliary code and outputs the auxiliary code.

Claim 16 (Original): A lossless coder for a digital signal in a floating-point format according to Claim 12, further comprising:

a down-sampling part that down-samples said first digital signal sample or said second digital signal sample to make the sampling frequency of the digital signal sample used for producing said first code sequence lower than the sampling frequency of said first digital signal sample; and

an up-sampling part that up-samples the digital signal sample used for producing said first code sequence to convert the sampling frequency thereof to the same sampling frequency as said first digital signal sample, thereby producing a digital signal sample used for producing said difference signal in the floating-point format.

Claim 17 (Original): A lossless coder for a digital signal in a floating-point format according to Claim 12, further comprising:

a precision adjusting part that converts the quantization precision of each first digital signal sample or each second digital signal sample to make the quantization precision of the digital signal sample used for producing said first code sequence lower than the quantization precision of said first digital signal sample; and

a reverse precision adjusting part that raises the quantization precision of the digital signal sample used for producing said first code sequence to the quantization precision of said first digital signal sample, thereby producing a digital signal sample used for producing said difference signal.

Claim 18 (Original): A decoder for a digital signal in a floating-point format, comprising:

a first expansion part that receives a first code sequence and decodes and expands the first code sequence to produce a first digital signal sample in an integer format;

a floating point formatting part that receives said first digital signal sample in the integer format and converts the first digital signal sample into a second digital signal sample in the floating-point format;

a combining part that receives a difference signal based on difference information and combines said difference signal and said first digital signal sample in the floating-point format to produce a third digital signal sample in the floating-point format.

Claim 19 (Original): A decoder for a digital signal in a floating-point format according to Claim 18, further comprising:

a second expansion part that decodes and expands said difference information to produce said difference signal.

Claim 20 (Original): A decoder for a digital signal in a floating-point format according to Claim 18 or 19, further comprising:

an auxiliary coding part that decodes an auxiliary code to produce adjustment information; and

a digit adjusting part that adjusts the digit of said first digital signal sample in the integer format or said combined signal based on said adjustment information.

Claim 21 (Original): A decoder for a digital signal in a floating-point format according to Claim 18 or 19, further comprising:

an up-sampling part that up-samples said first digital sample in the integer format to make the sampling frequency thereof equal to the sampling frequency of said difference signal, before converting said first digital sample into said second digital signal sample in the floating-point format.

Claim 22 (Original): A decoder for a digital signal in a floating-point format according to Claim 18 or 19, further comprising:

a precision adjusting part that converts the quantization precision of each first digital signal sample in the integer format to the quantization precision of said difference signal, before converting said first digital signal sample into said second digital signal sample in the floating-point format.

Claim 23 (Currently Amended): A coding program that makes a computer execute each step of a lossless coding method for a digital signal in a floating-point format according to ~~any one of Claims 1 to 6~~ Claim 1.

Claim 24 (Currently Amended): A decoding program that makes a computer execute each step of a decoding method for a digital signal in a floating-point format according to ~~any one of Claims 7 to 11~~ Claim 7.